

Deccan Education Society’s

FERGUSSON COLLEGE (AUTONOMOUS), PUNE – 4

**Department of Computer Science**

**A**

**Project Report**

**On**

**Matchmaking WebApp**

**In partial fulfillment of requirements of the completion of**

**M.Sc. (DS) - II**

**Semester – III**

**SUBMITTED BY:**

**Trupti Manoj Kachare (Roll no. 11118)**

**Under the Guidance of**

**Mrs. Swati Satpute**

**(2020 – 2021)**

**CERTIFICATE**

This is to certify that the project entitled **Matchmaking WebApp** submitted by

1. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

in partial fulfillment of the requirement of the completion of M.Sc. (Data Science)-II [Semester-III], has been carried out by them under our guidance satisfactorily during the academic year 2020-2021.

Place: Pune

Date: / /2021

**Head of Department**

**Department of Computer Science**

**Fergusson College, Pune**

**Project Guide:**

1. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Examiners Name Sign**

**1. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**2. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**AKNOWLEDGEMENTS**

We would sincerely like to thank our guide **Mrs. Swati Satpute** for her support, sincere guidance, timely help and valuable suggestions without which out project would have been impossible. We would also like to express our deepest gratitude to Computer Science Department **HOD Dr. Kavita Khobragade** ma’am and all those who have directly or indirectly helped us in completing this project.

Trupti Kachare 11118

[1. INTRODUCTION 5](#_Toc73138955)

[**1.1.** **PROJECT DESCRIPTION** 5](#_Toc73138956)

[**1.2.** **PROBLEM STATEMENT** 5](#_Toc73138957)

[2. DATA 6](#_Toc73138958)

[**2.1.** **DATA SOURCE** 6](#_Toc73138959)

[**2.2.** **DATA DESCRIPTION** 6](#_Toc73138960)

[**2.3.** **DATA FLOW** 7](#_Toc73138961)

[**2.4.** **DATA PREPROCESSING** 8](#_Toc73138962)

[3. VISUAL EXPLORATION 9](#_Toc73138963)

[**3.1.** **TYPES OF VISUALIZATION** 9](#_Toc73138964)

[**3.2.** **TECHNOLOGIES USED FOR VISUALIZATION** 9](#_Toc73138965)

[4. MODEL BUILDING 10](#_Toc73138966)

[**4.1.** **OBJECTIVE** 10](#_Toc73138967)

[**4.2.** **ALGORITHMS USED** 10](#_Toc73138968)

[**4.3.** **PERFORMANCE METRICS** 11](#_Toc73138969)

[5. SCREENSHOTS 12](#_Toc73138970)

[6. ANALYSIS 16](#_Toc73138971)

[7. FUTURE ENHANCEMENT 17](#_Toc73138972)

[8. BIBLIOGRAPHY 18](#_Toc73138974)

# INTRODUCTION

## **PROJECT DESCRIPTION**

Nowadays there are a lot of dating apps to help people to find their significant others. Some dating apps are quite popular among young people also there are some dating apps especially for the elderly. The purpose of those apps is to filter potential matches based on the users’ personal preferences, such as height, degree, habit, region, and occupation, etc.

Dating apps have been beneficial for many people who have fallen in love through those apps. But many people have bad experience regarding the matches on the app etc. So, using machine learning we can enhance the dating app experience.

## **PROBLEM STATEMENT**

With the help of Machine learning, we can improve the overall experience of dating/matchmaking apps to reduce the amount of time spent swiping. We could use AI to find users that are like each other and recommend dating profiles the same way Netflix recommends different movies. These AI systems like Netflix, Amazon etc. have been used in past. Similarly, we will use such AI systems for our dating app.

This type of dating app will make the tedious process of finding a partner online with fun. With machine learning, profiles can be potentially be clustered together with other similar profiles. This will reduce the time-consuming process of going through the non-compatible profiles.

# DATA

## **DATA SOURCE**

Data was taken from the following source:

<https://www.hackerearth.com/challenges/competitive/hackerearth-machine-learning-challenge-predict-match-percentage/>

## **DATA DESCRIPTION**

|  |  |
| --- | --- |
| Column Name | Description |
| User ID | Specifies the number/login id of the user |
| Username | Specifies the username |
| Age | Specifies the Age of the user |
| Status | Specifies the relationship status |
| Sex | Specifies the sex of the user |
| Orientation | Tells us whether the user is straight, bisexual or gay. |
| Drinks | Specifies the frequency of users’ drinking |
| Drugs | Specifies whether the user is having drugs |
| Height | Specifies the height of the user |
| Job | Specifies the users’ job |
| Location | Specifies the location where the user is currently staying. |
| Pets | Specifies whether the user likes pets or not. |
| Smokes | Specifies whether the user smokes or not |
| Body profile | Specifies the body profile of the user (fit, athletic etc.) |
| Education Level | Specifies the users’ education. |
| Dropped out | Specifies whether the user has dropped out of the college |
| Bio | Specifies short information about the user, generally hobbies, nature of that user etc. |
| Interests | Specifies the interests of the user. |
| Location Preference | Specifies the location preference of the user |

## **DATA FLOW**

Data Life Cycle

Timeline

Description automatically generated

New Data Point Life Cycle

Diagram

Description automatically generated

## **DATA PREPROCESSING**

Once the basic data cleaning, importing libraries is done we start clustering of data. Each profile belongs to a specific cluster number of groups. We successfully obtained the clustered groups into the data frame. With the clustered profile data, we sorted profiles based on the similarity. Clustering helps us to answer questions and find more interesting insights.

Natural Language Processing (NLP) – For some columns like bios or hobbies, there is a need to extract useful information from textual data. We can use NLP based on this textual data and combine them with algorithms to get more accurate matches. NLP can also be useful in different aspects such as – trending groups of people, top 10 matches etc.

The recommendations system is good for matchmaking. Not only it can recommend good matches according to your basic preferences but also, we can analyze their user behavior.

**PREPROCESSING**

The data has too many categorical columns to handle. For actual model building, a smaller number of columns are used. The main concern was to reduce the number of columns.

Chart, histogram

Description automatically generatedChart, bar chart, histogram

Description automatically generated

Before After

Chart, bar chart, histogram

Description automatically generatedChart, bar chart

Description automatically generated

Before After

As shown above we tried to reduce the number of distinct values present in all columns by using count plot (we kept values with high frequency count) and we merged similar categories using domain knowledge (like for drinks column we merged very often and desperately to often).

All this because in the end we were going to apply one hot encoding to the data so blindly applying one hot encoding to the data would create very large number of features so to avoid that the preprocessing was done in above manner.

**NLP PREPROCESSING**

Load the necessary libraries as well as libraries from NLTK. The NLTK library will be the primary library we use to process the text data from the dating profile bios. Following are the preprocessing steps:

1. **Tokenizing the user bios** - Tokenization is the process of splitting up sentences into individual tokens. A function is created for tokenization, in order to have cleaned bio. Within the function -
   1. Create a library of stop words (the, a, of etc.) to exclude these words which do not make any sense.
   2. Lowercase the entire string or sentence
   3. Replace the punctuations.
   4. Split the string or sentence on the spaces between each word.
   5. Iterate through each word in the sentence then lemmatize the word or remove if it is a stop word.
2. **Lemmatizing** – Breaking down the words to their base form of the word, lemmatization is the best option.
3. **Bigrams** - implement some **bigrams** (pairs of words). We create a list of pairs of words and their respective frequency scores.

# 

‘Knowing each other’

'Since start I knew you wanted to talk something, there was some burden like thing you are carrying within you. You say freely I would listen to it, don’t hold any hesitations, trust me.'

Sample of user’s bio value in raw data After NLP modelling that would be converted into this label.

Likewise, we transformed all the bio values into 6 relevant labels and applied one hot encoding on that too.

# 

# VISUAL EXPLORATION

## **TYPES OF VISUALIZATION**

1. **BAR PLOT** – This graph is used for studying the count of the age, height, education level i.e. the numeric columns as well as for the categorical columns.

For NLP, the most frequent words, unique words and their count is plotted using bar plot.

1. **BOX PLOT** - This graph is used for detecting any outliers in the dataset.
2. **LINE GRAPH** – For looking at the model performance, line graph is used.
3. **BUBBLE CHART** - Bubble Charts are typically used to [compare](https://datavizcatalogue.com/search/comparisons.html) and show the [relationships](https://datavizcatalogue.com/search/relationships.html) between categorized circles, by the use of positioning and [proportions](https://datavizcatalogue.com/search/proportions.html). The overall picture of Bubble Charts can be used to analyze for [patterns/correlations](https://datavizcatalogue.com/search/patterns.html).

## **TECHNOLOGIES USED FOR VISUALIZATION**

Python libraries like, Matplotlib, Seaborn are used for visualization.

For dashboard, Tableau is used.

# MODEL BUILDING

## **OBJECTIVE**

Clustering is used to cluster the dating profiles with one another. By doing so, the users will get the matches like themselves, instead of the profiles unlike their own. With machine learning, profiles can potentially be clustered together with other similar profiles. This will reduce the number of profiles that are not compatible with one another. From these clusters, users can find other users more like them.

## **ALGORITHMS USED**

The dating app is built using algorithms of Machine Learning. More specifically, utilizing unsupervised machine learning in the form of clustering is the main objective. Algorithms used are:

**UNSUPERVISED**

**K-Means Clustering** – This is one of the simplest and popular unsupervised machine learning algorithms. Typically, unsupervised algorithms make inferences from datasets using only input vectors without referring to known, or labelled, outcomes. You’ll define a target number k, which refers to the number of centroids you need in the dataset. A centroid is the imaginary or real location representing the center of the cluster. The K-means algorithm identifies k number of centroids, and then allocates every data point to the nearest cluster, while keeping the centroids as small as possible. The ‘means’ in the K-means refers to averaging of the data; that is, finding the centroid.

**SUPERVISED**

**Logistic Regression** – This is a classification algorithm used to assign observations to a discrete set of classes. It is a predictive analysis algorithm and based on the concept of probability. The hypothesis of logistic regression tends it to limit the cost function between 0 and 1.

## **PERFORMANCE METRICS**

For various algorithms, depending on supervised and unsupervised learning different performance metrics are used.

**UNSUPERVISED**

**WCSS –**

Chart, line chart

Description automatically generated There is slight edge for number of clusters 4. So we decided to analyze performance of 4 and 5 clusters.

**Silhouette Score** (Higher the better) -

Chart, line chart

Description automatically generated

Silhouette score of number 4 is higher than 5 so it tells number of clusters should be 4.

Chart

Description automatically generatedChart

Description automatically generated

Clusters in left side plot looks well distributed and leakage(color spill towards left side) is also less and in addition to this volume of each color(which represent number of users in that cluster) looks well distribute in left plot. So this confirms the number of clusters to be 4.

**SUPERVISED**

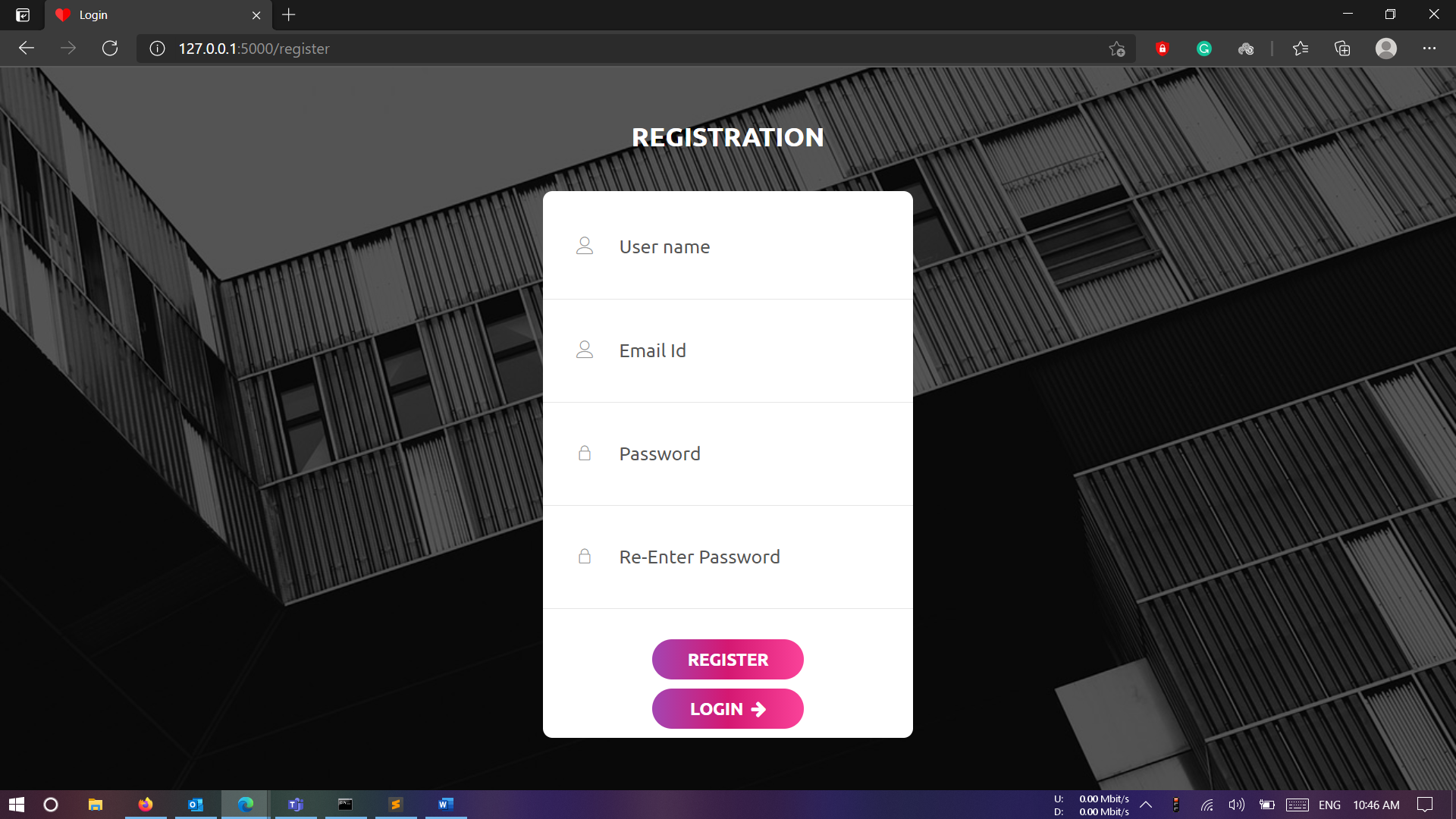
**Confusion Matrix** –

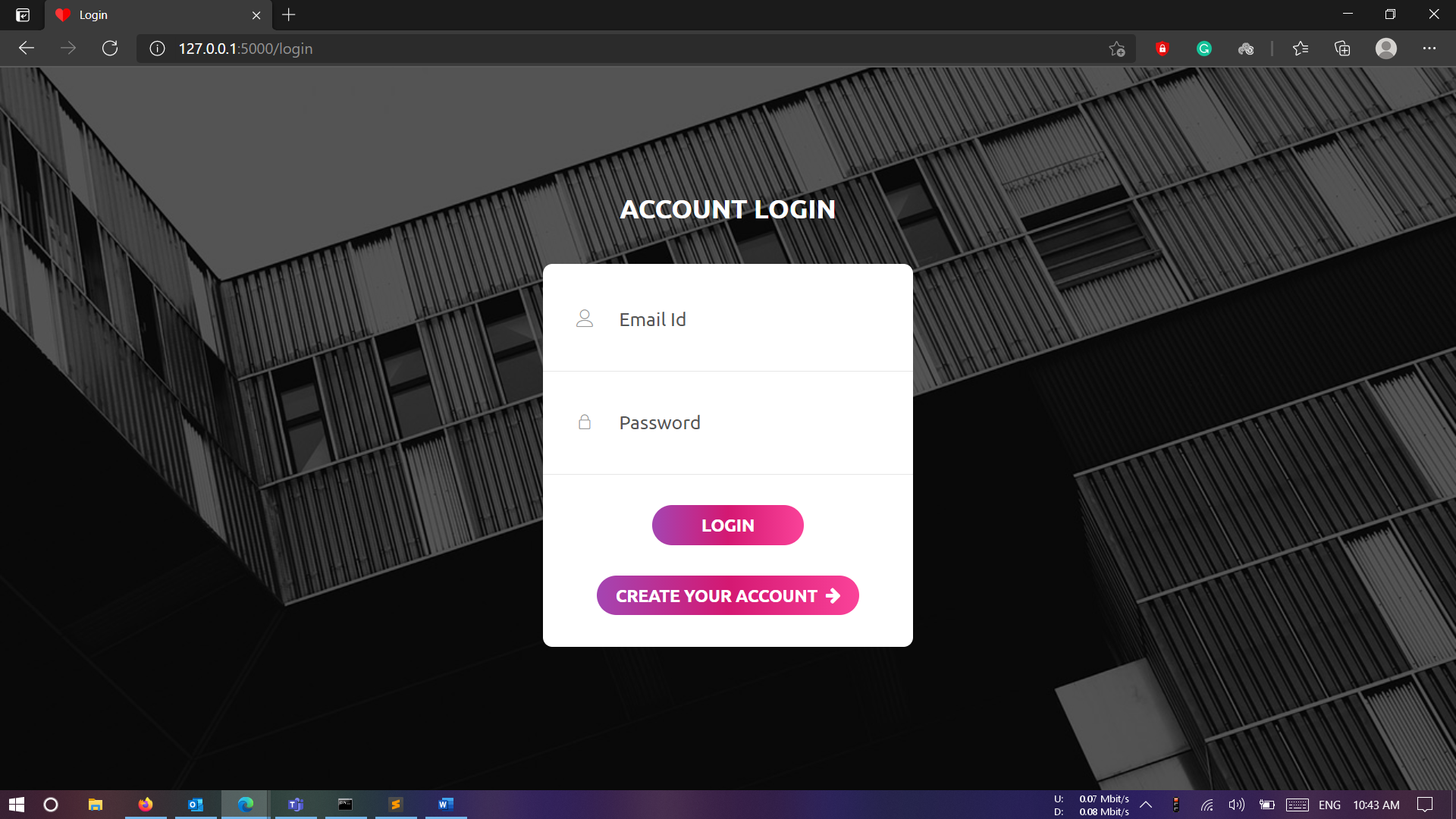
A picture containing calendar

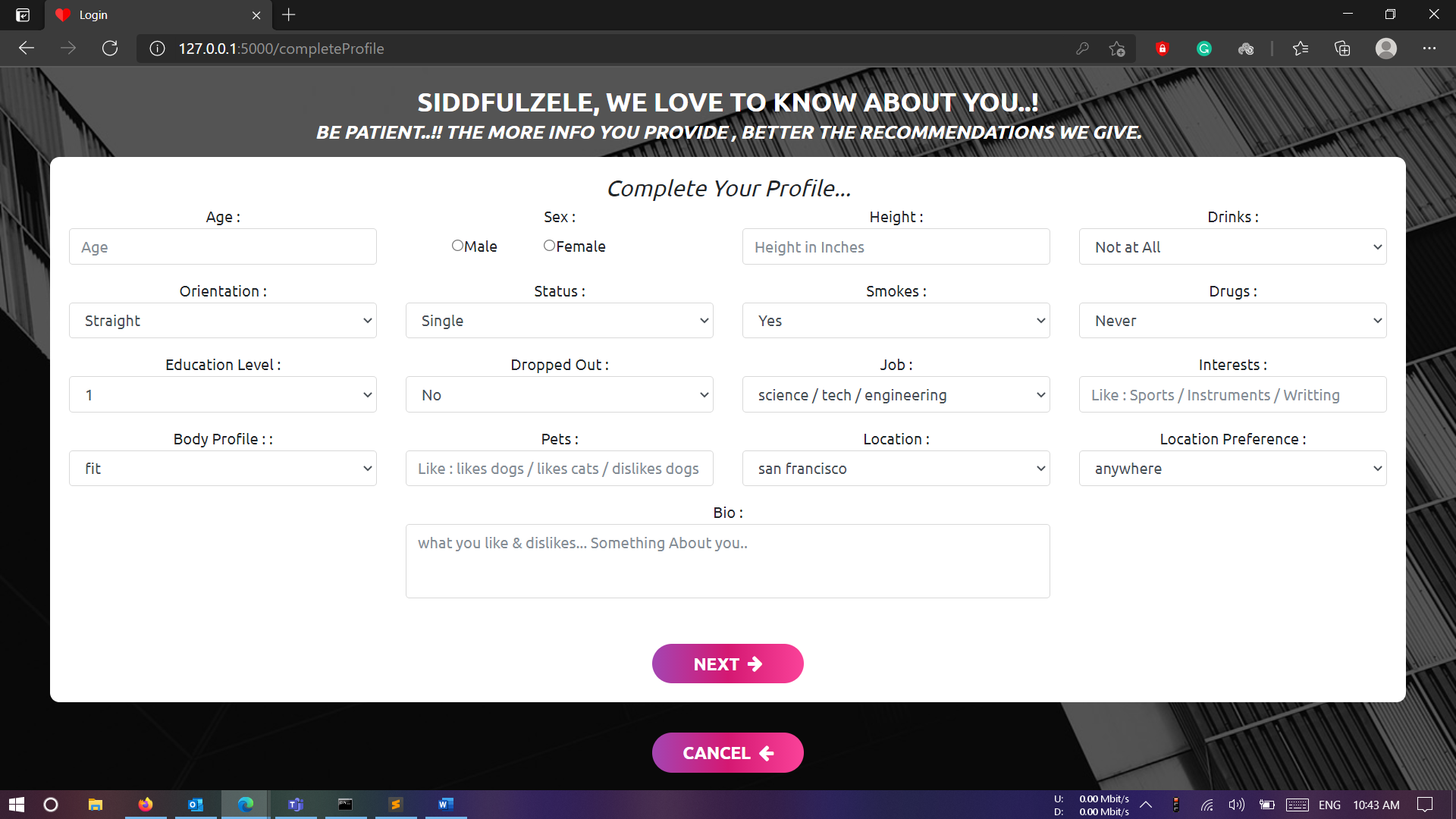
Description automatically generated

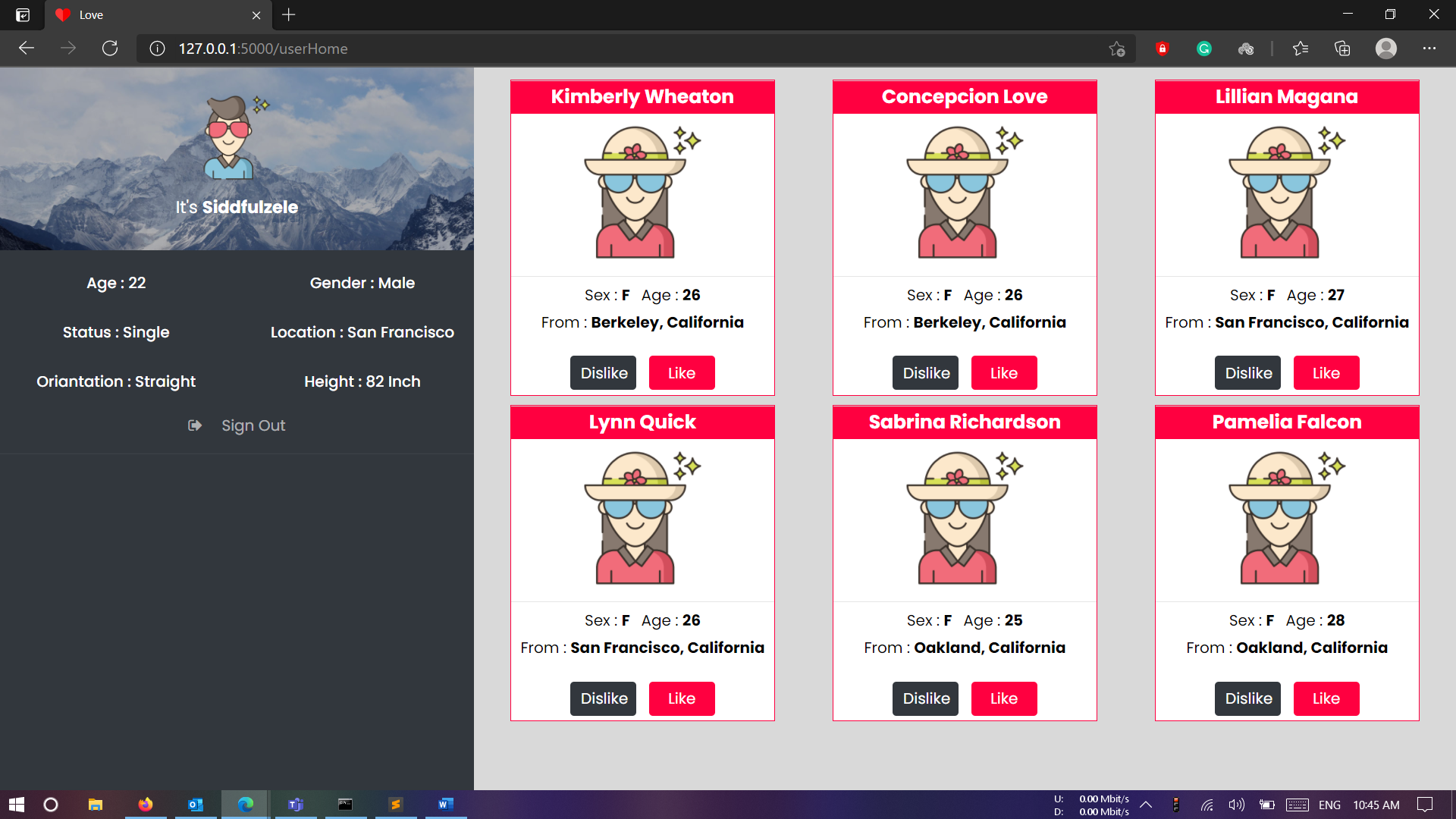
Above is the confusion matrix of logistic model’s output on test split. We can see 0 misclassification this is due to the kind of data we provided to model was having values either 0 or 1. So the data was linearly separable, and model could find it easy to classify samples as no continues values were present in data.

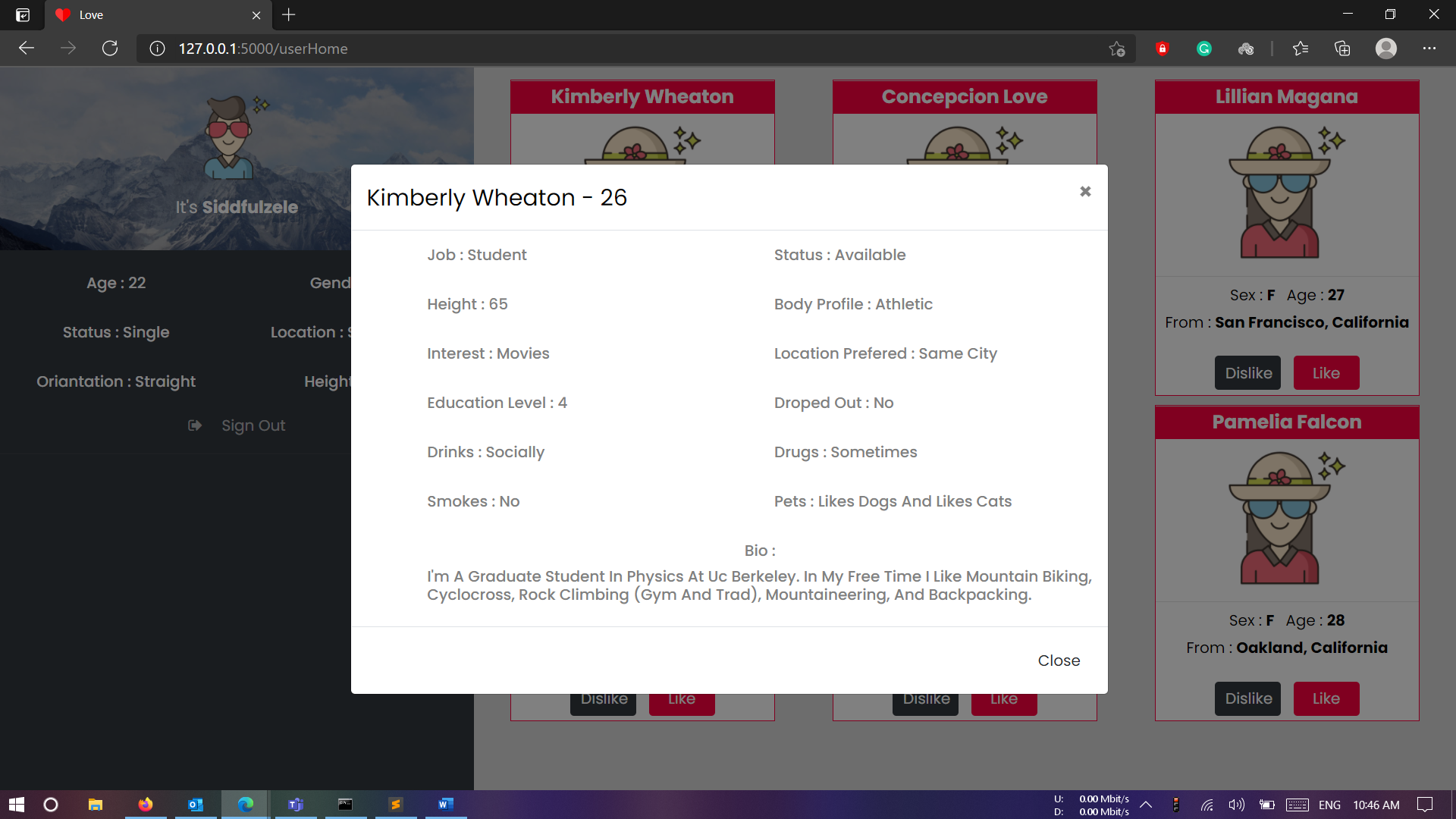
# SCREENSHOTS

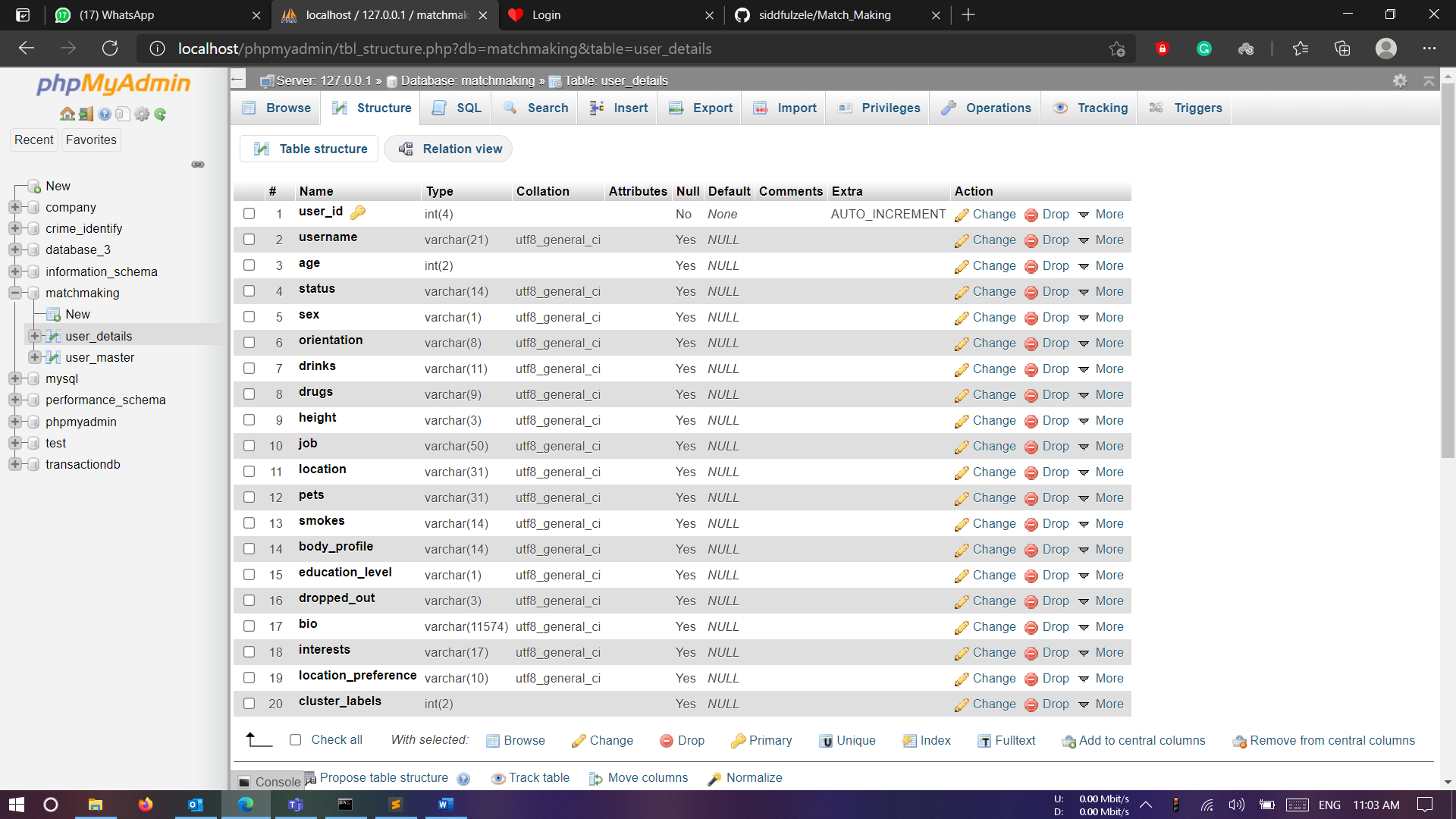


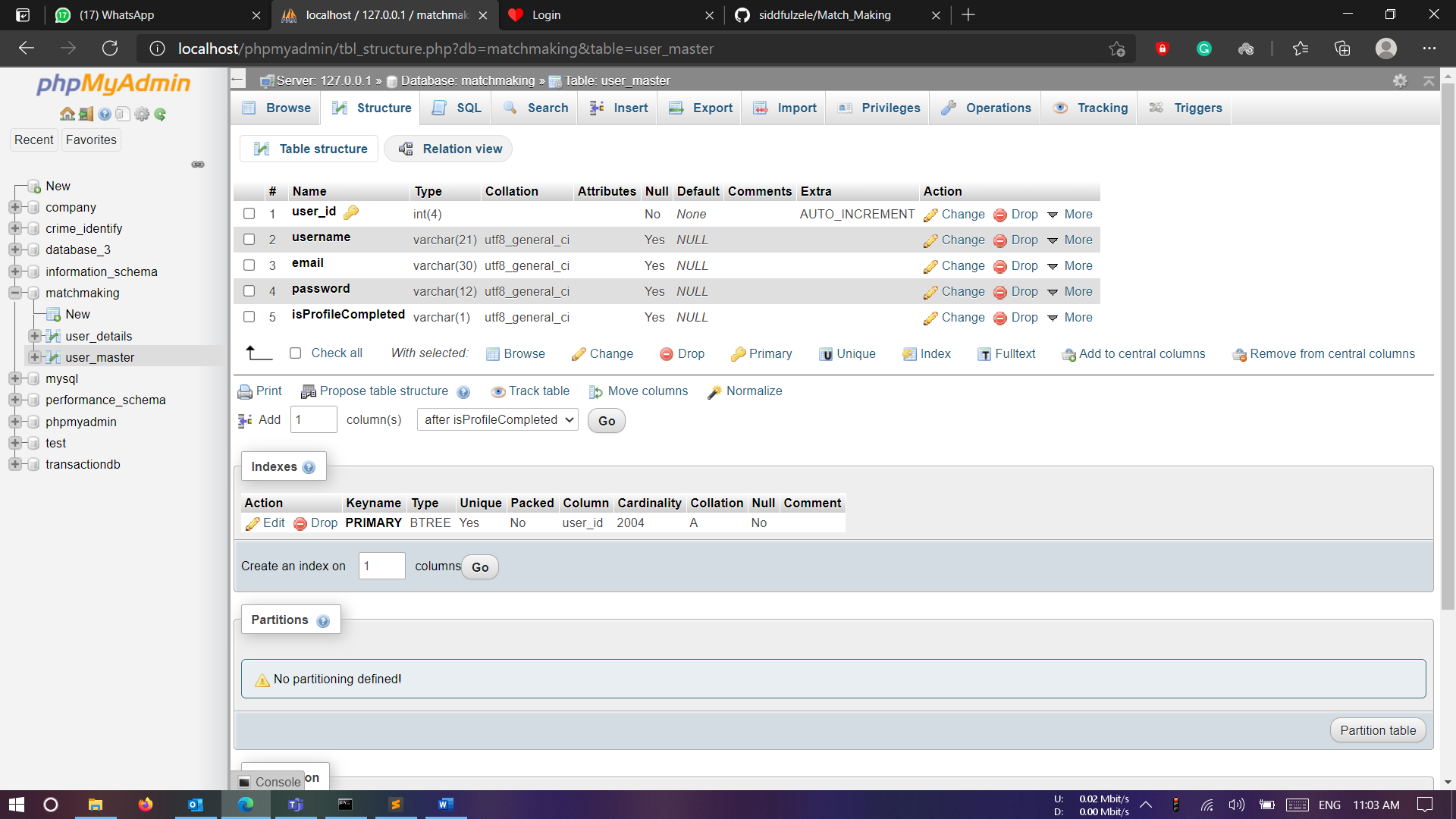












# ANALYSIS

Chart, histogram

Description automatically generated

Chart, bubble chart

Description automatically generated

Published on Tableau Public - <https://public.tableau.com/app/profile/trupti5700/viz/MatchMaking_Dashboard/Dashboard1>

# FUTURE ENHANCEMENT

# To conclude, Matchmaking WebApp overcomes many limitations. Following are the advantages of this app like,

* 1. Easy implementation environment.
  2. Spend less time swiping
  3. Recommendations based on personal interests

**Scope for future development**

There are other potential improvements to be made in this project. Following are the future scope for the project.

* 1. Implementing a way to include a chat facility to the recommended users.
  2. Users will also be able to upload their own photo, instead of the animations provided right now.
  3. Users can also link social media accounts to their profiles.

# BIBLIOGRAPHY

* 1. Sklearn Documentation for Algorithms - <https://scikit-learn.org/stable/unsupervised_learning.html>
  2. Documentations for Visualization
     1. Matplotlib - <https://matplotlib.org/>
     2. Seaborn - <https://seaborn.pydata.org/>
     3. Word Cloud for NLP - <https://pypi.org/project/wordcloud/>
  3. Pandas Documentation - <https://pandas.pydata.org/docs/user_guide/index.html>